AMENDMENTS TO THE SPECIFICATION

On page 15, please replace the paragraph starting at line 31 with the following paragraph:

The base data structure for the block specifies the generic fields and interfaces that need to be supported by a block. Some of the methods are purely virtual and have no specific implementation in the base block class. In order to define a specific block (such as an Integrator block), one needs to subclass the base block class and provide explicit definitions for these virtual methods. An example of the subclassing of a block may be seen by examining an Integrator block. **FIG. 3** depicts the desired behavior of an Integrator block 30. In order to create the subclass, four major categories of information within the subclass must be specified, the block parameters, the methods used in editing, the methods used in compilation, and the methods used in execution. The elemental dynamic system embodied by the block may be parameterized as illustrated in **FIGS. 1A-1C.** Each block needs to be able to specify its list of expected parameters. The block diagram editor's Attribute-Editing tool may allow users to specify the parameters for the block when they use it in their models. In the Integrator block example, the block has one parameter that specifies the block's initial condition for the block's state. Regarding the methods used in editing, the subclass needs to specify a method that renders its icon. For example, the Integrator block may implement a method that makes its icon be a box with a "1/s" within the box. Also, the subclass needs to instantiate a method that allows access of the block parameters from the GUI's Attribute-Editing tool. For the Integrator example, this method would allow users to specify the Initial Condition parameter on a GUI for the block. For the methods used in compilation, the subclass needs to instantiate methods that help in the compilation of the block diagram model in which it is placed. These methods help specify the compiled information for the inputs and outputs of the block. For instance, the Integrator block may specify a method that ensures that if the input to the Integrator is a vector, then the output is a vector of the same size. For methods used in execution, the subclass needs to instantiate specific Output, Derivative, and Update methods that represent the block behavior. In the case of the Integrator block, ~~an~~ Output and Derivative methods are needed. It should be noted that in Simulink® the Integrator block has additional methods that are not illustrated here. The Output method sets the output to be equal to the state. The Derivative method sets the derivative of the state to be equal to the input.

On page 39, please replace the paragraph starting at line 4 with the following paragraph:

Information related to the compiled block diagram may be presented to users in an automatically generated report. This report allows users to quickly obtain documentation of the functional description of their model. Information related to the execution of a particular model (such ~~at~~ as the time taken to execute various portions of the model and the coverage of various portions of the model) may be obtained automatically and presented to the user as a report.

On page 46, please replace the paragraph starting at line 27 with the following paragraph:

In accordance with another embodiment of the present invention, a medium holding computer executable instructions ~~steps~~ for carrying out a method of altering a graphical model is provided. The method includes identifying a component of the graphical model for conversion. The component is processed to identify one of similarities with other components and similarities with selected characteristics, and automatically convert the component into a reference.